

### **Amendments to the Claims**

Please amend Claims 1, 6, 13, 30, 35, 42 and 50, all as shown below. Applicant respectfully reserves the right to prosecute any originally presented or canceled claims in a continuing or future application.

1. (Currently Amended)      A method for supporting a portal application, comprising:
- accepting a request, at a container on one or more web servers, from a user that interacts with a graphical user interface (GUI) of a web application at a client side;
  - mapping the request to a control tree factory ~~to generate a control tree~~, wherein the control tree factory can obtain an XML stream from different sources and parse the XML stream into a Document Object Model (DOM) tree ~~is independent of the container and is accessible from other containers, wherein at least one of the containers is associated with at least one of a different protocol and a different application framework from the container;~~
  - processing the DOM tree and generating [[the]] a control tree in the container by the control tree factory from the DOM tree based on the request, wherein the control tree is a logical representation of the graphical user interface (GUI), wherein the control tree includes a set of controls, ~~that are related hierarchically to each other, and wherein each control of the set of controls~~ each of which represents ~~one or more corresponding~~ at least one of a graphical element and a functional element~~[[s]] in the GUI of the web application;~~
  - advancing the control tree through at least one lifecycle stage in a sequence of one or more lifecycles, wherein at least one control in the control tree operates to interact with another control in the control tree through an event notification mechanism;
  - aggregating the output of each control of the set of controls in the control tree to produce a response based on the request;
  - ~~providing the response to the container that contains the control tree; and~~
  - ~~providing the response to the GUI of the web application at the client side.~~

2-3. (Canceled).

4. (Previously Presented)      The method of claim 1 wherein the step of generating a control tree comprises:
- creating a metadata representation of a control tree; and

generating a class to construct the control tree based on the metadata representation.

5. (Original) The method of claim 1 wherein:
  - the request is a hypertext transfer protocol request (HTTP); and
  - the request originates from a web browser.
6. (Currently Amended) The method of claim 1, further comprising:
  - providing the response to a web browser.
7. (Original) The method of claim 1 wherein:
  - the control tree is advanced through the at least one lifecycle stage by an interchangeable lifecycle component.
8. (Original) The method of claim 1 wherein:
  - each one of the set of controls can have an interchangeable persistence mechanism.
9. (Original) The method of claim 1 wherein:
  - each one of the set of controls can render itself according to a theme.
10. (Canceled).
11. (Original) The method of claim 1 wherein:
  - one of the set of controls can advance through the series of at least one lifecycle stage in parallel with another of the controls.
12. (Original) The method of claim 1 wherein:
  - a lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose.
13. (Currently Amended) The method of claim 1 wherein:
  - the response is a hypertext transfer protocol (HTTP) response.
14. (Original) The method of claim 1 wherein:
  - controls can raise events and respond to events.

15. (Original) The method of claim 1 wherein:

each one of the set of controls can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.

16-29. (Canceled).

30. (Currently Amended) A machine readable storage medium having instructions stored thereon that when executed by a processor cause a system to:

accept a request, at a container on one or more web servers, from a user that interacts with a graphical user interface (GUI) of a web application ~~at client side;~~

map the request to a control tree factory ~~to generate a control tree~~, wherein the control tree factory can obtain an XML stream from different sources and parse the XML stream into a Document Object Model (DOM) tree ~~is independent of the container and is accessible from other containers, wherein at least one of the other containers is associated with at least one of a different protocol and a different application framework from the container;~~

process the DOM tree and generate a ~~[[the]]~~ control tree in the container by the control tree factory from the DOM tree based on the request, wherein the control tree is a logical representation of the graphical user interface (GUI), wherein the control tree includes a set of controls, ~~that are related hierarchically to each other, and wherein each control of the set of controls~~ each of which represents ~~one or more corresponding graphical and functional~~ at least one of a graphical element and a functional element~~[[s]]~~ in the GUI ~~of the web application;~~

advance the control tree through at least one lifecycle stage in a sequence of one or more lifecycles, wherein at least one control in the control tree operates to interact with ~~another control in the control tree through an event notification mechanism;~~ and produce a response based on the request;

~~aggregate the output of each control of the set of controls in the control tree to produce a response based on the request;~~

~~provide the response to the container that contains the control tree; and~~

aggregate the output of each of the set of controls and provide the response to the GUI of the web application at the client side.

31. (Canceled).

32. (Canceled).

33. (Previously Presented) The machine readable medium of claim 30, further comprising instructions that when executed cause the system to:

create a metadata representation of the control tree; and  
generate a class to construct the control tree based on the metadata representation.

34. (Original) The machine readable medium of claim 30 wherein:  
the request is a hypertext transfer protocol request (HTTP); and  
the request originates from a web browser.

35. (Currently Amended) The machine readable medium of claim ~~[[32]]~~ 30, further comprising instructions that when executed cause the system to:  
provide the response to a web browser.

36. (Original) The machine readable medium of claim 30 wherein:  
the control tree is advanced through the at least one lifecycle stage by an interchangeable lifecycle component.

37. (Original) The machine readable medium of claim 30 wherein:  
each one of the set of controls can have an interchangeable persistence mechanism.

38. (Original) The machine readable medium of claim 30 wherein:  
each one of the set of controls can render itself according to a theme.

39. (Canceled).

40. (Original) The machine readable medium of claim 30 wherein:  
one of the set of controls can advance through the series of at least one lifecycle stage in parallel with another of the controls.

41. (Original) The machine readable medium of claim 30 wherein:

a lifecycle stage is one of: init, load state, create child controls, load, raise events, pre-render, render, save state, unload and dispose.

42. (Currently Amended) The machine readable medium of claim [[32]] 30 wherein:  
the response is [[an]] a hypertext transfer protocol (HTTP) response.

43. (Original) The machine readable medium of claim 30 wherein:  
controls can raise events and respond to events.

44. (Original) The machine readable medium of claim 30 wherein:  
each one of the set of controls can be one of: Book, Page, Window, Menu, Layout, Portlet, Theme, Placeholder, Shell, LookAndFeel, Desktop, Body, Footer, Header, Head, Titlebar, ToggleButton, TreeView, TreeViewWithRadioButtons.

45-49. (Canceled).

50. (Currently Amended) The method of claim 1, wherein:  
the one or more lifecycles of the control tree is provided and managed by the container and can be modified by the container, and one or more controls in the control tree can save their states at a particular stage in the one or more lifecycles and can then reload these states at a later stage in the one or more lifecycles.

51. (Previously Presented) The method of claim 1, wherein:  
each container associates a context object with the control tree factory, wherein each context object provide access to the protocol and application framework that is associated with that container.

52. (Previously Presented) The method of claim 1, wherein:  
the control tree factory uses one or more meta data to construct statically created controls at initialization of the control tree, wherein dynamically created controls are created in the control tree in reaction to state, context, and events during a control tree lifecycle.